COM S/CPR E 513X: Foundations and Applications of Program Analysis

Iowa State University
Spring 2016

Instructor: Wei Le (weile@iastate.edu)
Class schedule: T Th 11:00am-12:15pm, GILMAN 2354
Office hours: Th 2:00pm-3:00pm ATANASOFF 210

1 Course Description
Program analysis aims to automatically reason about the code and program executions for predicting software behaviors. The course will teach how to determine program properties (foundational knowledge of program analysis) and what are the purposes to do so (applications of program analysis). The course will teach concepts, algorithms, tools, benchmarks and methodologies needed for solving problems using program analysis and for establishing foundations for program analysis research.

2 Course Objectives
After successfully completing this course, students will be able to:

• know terminologies and mathematical frameworks needed to read program analysis literature,
• know the capabilities and tradeoffs of program analysis algorithms,
• implement classical program analysis algorithms,
• gain problem solving skills and apply program analysis to solve software engineering and security problems, and
• start program analysis research

3 Prerequisites
• COM S 331: Theory of Computation
• COM S 342: Principles of Programming Languages

4 Textbooks and Other Resources
Text books (they are not required but highly recommended):


**Program analysis courses** taught by other instructors:

1. Alex Aiken, Stanford, [CS 357 Techniques for Program Analysis and Verification](#)
2. Monica Lam, Stanford, [CS243 Program Analysis and Optimization](#)
4. Jens Palsberg, UCLA, [CS232 Static Program Analysis](#)
5. Mayur Naik, Georgia Tech, [CS6340 Software Analysis and Testing](#)
6. Stephen Chong, Harvard, [CS252r Advanced Topics in Programming Languages](#)
7. Mooly Sagiv, Tel Aviv University, [Program Analysis](#)
8. Evan Chang, University of Colorado Boulder, [CSCI7135 Program Analysis: Theory and Practice](#)

5 **Course Work and Evaluation**

- **Reading and survey (20%)**: You are expected to read assigned papers and participate in paper discussions in class. Some topics and papers see below. You are also expected to select a topic of interest and write a survey of the topic.
  1. Dataflow: infeasible paths (Bodik)
  2. Points-to analysis: Anderson-Style
  3. Slicing: static slicing (Weiser)
  4. Symbolic execution: KLEE (Cadar)
  5. Interprocedural analysis: IFDS (Reps)
  6. Program analysis application: ESP (Das)
  7. Dynamic analysis: Daikon (Ernst)

- **Implementation (40%)**: You will implement a set of program analysis algorithms. Some algorithms see below.
  1. Program representation: Call Graph, CFG, ICFG
  2. Dataflow analysis
  3. PDG and slicing, tainting analysis
  4. Points-to analysis

- **Research project (40%)**: You will select a small problem of interest and apply program analysis to solve it. The submissions include a proposal, a mid-point check presentation, a final report and a demo.
6 Topics

1. Overview
2. Points-to analysis
3. Computability, complexity, soundness, completeness, flow-sensitivity, context-sensitivity, path-sensitivity
4. Control flow analysis and type inferences: call graphs, cfg, icfg
5. Dependency and program slicing, chopping, taint analysis
6. Abstract interpretation
7. Dataflow analysis
8. Interprocedural analysis
9. Symbolic execution
10. Infeasible paths detection
11. Typestate
12. Delta-debugging
13. Program analysis on security
14. Dynamic analysis
15. Hybrid analysis
16. Program analysis and its related areas
17. Experimentation

7 Course Policies

**Absence and late homework policy:** We do not grade late homework. Please submit it in time. Your attendance of the class will be considered as a part of *Class Participation* grade. If you are too sick to attend the class or finish the homework, please send in an email to explain the situation.

**Iowa State University’s policy on academic dishonesty:** Suspected academic misconduct will be reported to the dean of students office http://www.dso.iastate.edu/ja/academic/misconduct.html